Appendix

A Proofs

In Appendix A, we provide complete proofs of the theoretical results.

A.1 Proof of Theorem 1

Proof. We just need to show that $\tilde{g}$ is an unbiased estimator of a sub-gradient of $L_q(\theta)$ at θ^t, namely $\mathbb{E}\tilde{g} \in \partial L_q(\theta^t)$.

At first, it holds that

$$\mathbb{E}\tilde{g}^t = \frac{1}{q}\mathbb{E}\sum_{i\in Q} g_i^t + g_R^t = \frac{1}{q}\sum_{i=1}^{n} P(i \in Q) g_i^t + g_R^t = \frac{1}{q}\sum_{j=1}^{n} P((j) \in Q) g_{(j)}^t + g_R^t \ ,$$

where $g_i^t \in \partial L_i(\theta^t)$ is a sub-gradient of L_i at θ^t and $g_R^t \in \partial R(\theta^t)$. In the above equality chain, the third equality is simply the definition of expectation, and the last equality is because $((1),(2),\ldots,(n))$ is a permutation of $(1,2,\ldots,n)$.

For any given index j, define $A_j = ((1),(2),\ldots,(j-1))$, then

$$\begin{aligned} P((j) \in Q) &= P\left((j) \in \text{q-argmax}_{i\in S} L_i(\theta)\right) \\ &= P\left((j) \in S \text{ and } S \text{ contains at most } q-1 \text{ items in } A_j\right) \\ &= P\left((j) \in S\right) P\left(S \text{ contains at most } q-1 \text{ items in } A_j | (j) \in S\right) \\ &= P\left((j) \in S\right) \textstyle\sum_{l=0}^{q-1} P\left(S \text{ contains } l \text{ items in } A_j | (j) \in S\right). \end{aligned} \tag{6}$$

Notice that S is randomly chosen from sample index set $(1,2,\ldots,n)$ without replacement. There are in total $\binom{n}{s}$ different sets S such that $|S| = s$. Among them, there are $\binom{n-1}{s-1}$ different sets S which contains the index (j), thus

$$P\left((j) \in S\right) = \frac{\binom{n-1}{s-1}}{\binom{n}{s}} \ . \tag{7}$$

Given the condition $(j) \in S$, S contains l items in A_j means S contains $s-l-1$ items in $\{(j+1),(j+2)\ldots,(n)\}$, thus there are $\binom{j-1}{l}\binom{n-j}{s-l-1}$ such possible set S, whereby it holds that

$$P\left(S \text{ contains } l \text{ items in } A_j | (j) \in S\right) = \frac{\binom{j-1}{l}\binom{n-j}{s-l-1}}{\binom{n-1}{s-1}} \ . \tag{8}$$

Substituting Equations (7) and (8) into Equation (6), we arrive at

$$P((j) \in T) = \frac{\binom{n-1}{s-1}}{\binom{n}{s}} \sum_{l=0}^{q-1} \frac{\binom{j-1}{l}\binom{n-j}{s-l-1}}{\binom{n-1}{s-1}} = \frac{\sum_{l=0}^{q-1}\binom{j-1}{l}\binom{n-j}{s-l-1}}{\binom{n}{s}} = \gamma_j \ .$$

Therefore,

$$\mathbb{E}\tilde{g}^t = \frac{1}{q}\sum_{j=1}^{n} P((j) \in Q) g_{(j)}^t + g_R^t = \frac{1}{q}\sum_{j=1}^{n} \gamma_j g_{(j)}^t + g_R^t \in \partial L_q(\theta^t) \ ,$$

where the last inequality is due to the aditivity of sub-gradient (for both convex and weakly convex function) □

A.2 Proof of Proposition 1

We just need to show that

$$\lim_{j,n\to\infty, j/n=z} \gamma_j = \sum_{l=0}^{q-1} \frac{1}{n}\left(\frac{j}{n}\right)^l \left(\frac{n-j}{n}\right)^{s-l-1} \frac{s!}{l!(s-l-1)!} \ , \tag{9}$$

then we finish the proof by changing variable $z = \frac{j}{n}$.

At first, the Stirling's approximation yields that when n and j are both sufficiently large, it holds that

$$\binom{n}{j} \sim \sqrt{\frac{n}{2\pi j(n-j)}} \frac{n^n}{j^j (n-j)^{n-j}} \ . \tag{10}$$

Thus,

$$\lim_{j,n\to\infty, j/n=z} \frac{\binom{n-s}{j-1-l}}{\binom{n-1}{j-1}} = \frac{\frac{n^{n-s}}{j^{j-1-l}(n-j)^{n-j-s+1+l}}}{\frac{n^{n-1}}{j^{j-1}(n-j)^{n-j}}} = \frac{j^l (n-j)^{s-l-1}}{n^{s-1}} = \left(\frac{j}{n}\right)^l \left(\frac{n-j}{n}\right)^{s-l-1}, \tag{11}$$

where the first equality utilize Equation (10) and the fact that $s, l, 1$ are negligible in the limit case (except the exponent terms).

On the other hand, it holds by rearranging the factorial numbers that

$$\frac{1}{n} \frac{\binom{n-s}{j-1-l}}{\binom{n-1}{j-1}} \frac{s!}{l!(s-l-1)!} = \frac{\binom{j-1}{l}\binom{n-j}{s-l-1}}{\binom{n}{s}} \ . \tag{12}$$

Combining Equations (11) and (12) and summing l, we arrive at Equation (9).

By noticing $s > q$, it holds that

$$\begin{aligned}
\frac{d}{dz}\gamma(z) &= \sum_{l=1}^{q-1} l z^{l-1}(1-z)^{s-l-1} \frac{s!}{l!(s-l-1)!} - \sum_{l=0}^{q-1} (s-l-1) z^l (1-z)^{s-l-2} \frac{s!}{l!(s-l-1)!} \\
&= \sum_{l=1}^{q-1} z^{l-1}(1-z)^{s-l-1} \frac{s!}{(l-1)!(s-l-1)!} - \sum_{l=0}^{q-1} z^l (1-z)^{s-l-2} \frac{s!}{l!(s-l-2)!} \\
&= \sum_{l=0}^{q-2} z^l (1-z)^{s-l-2} \frac{s!}{l!(s-l-2)!} - \sum_{l=0}^{q-1} z^l (1-z)^{s-l-2} \frac{s!}{l!(s-l-2)!} \\
&= -z^{q-1}(1-z)^{s-q-1} \frac{s!}{l!(s-l-2)!} \\
&\propto -z^{q-1}(1-z)^{s-q-1}.
\end{aligned}$$

In other word, $1 - \frac{1}{s}\gamma(z)$ is the cumulative of $\mathrm{Beta}(q, s-q)$ when $n \to \infty$.

A.3 Proof of Theorem 2

Proof. Notice that $\tilde{g}^t$ is a sub-gradient of $L_Q(\theta^t)$ where $L_Q(\theta^t) = \frac{1}{q}\sum_{i\in Q} L_i(\theta^t) + R(\theta^t)$. Suppose $\tilde{g}^t = \frac{1}{q}\sum_{i\in Q} g_i(\theta^t) + g_R(\theta^t)$ where $g_i(\theta^t)$ is a sub-gradient of $L_i(\theta^t)$ and $g_R(\theta^t)$ is a sub-gradient of $R(\theta^t)$. Then

$$\|\tilde{g}^t\|^2 = \left\| \frac{1}{q}\sum_{i\in Q} g_i(\theta^t) + g_R(\theta^t) \right\|^2 \le 2\left(\left\| \frac{1}{q}\sum_{i\in Q} g_i(\theta^t) \right\|^2 + \left\| g_R(\theta^t) \right\|^2 \right) \le 2(G_1^2 + G_2^2) \ . \tag{13}$$

Meanwhile, it follows Theorem 1 that $\tilde{g}^t$ is an unbiased estimator of a sub-gradient of $L_q(\theta^t)$. Together with Equation (13), we obtain the statement (1) by the analysis of convex stochastic sub-gradient descent in Boyd and Mutapcic (2008).

Furthermore, suppose $L_i(\theta) + \frac{\rho}{2}\|\theta\|^2$ is convex for any i, then $L_q(\theta) + \frac{\rho}{2}\|\theta\|^2 = \frac{1}{q}\sum_{j=1}^n \gamma_j \left(L_{(j)}(\theta) + \frac{\rho}{2}\|\theta\|^2\right) + R(\theta)$ is also convex, whereby $L_q(\theta)$ is ρ-weakly convex. We obtain the statement (2) by substituting into Theorem 2.1 in Davis and Drusvyatskiy (2018). □

A.4 Proof of Theorem 3

Before proving Theorem 3, we first show the following proposition, which gives an upper bound for γ_j:

Proposition 2. *For any $j \in \{1, \dots, n\}$, $\gamma_j \leq \frac{s}{n}$.*

Proof. The value of γ_j is equal to the probability of ordered SGD choosing the j-th sample in the ordered sequence $(L_{(1)}(\theta; \mathcal{D}), \dots, L_{(n)}(\theta; \mathcal{D}))$, which is at most the probability of mini-batch SGD choosing the j-th sample. The probability of mini-batch SGD choosing the j-th sample is $\frac{s}{n}$. □

We are now ready to prove Theorem 3 by finding an upper bound on $\sup_{\theta \in \Theta} \mathbb{E}_{(x,y)}[\ell(f(x;\theta), y)] - L_q(\theta; \mathcal{D})$ based on McDiarmid's inequality.

Proof of Theorem 3. Define $\Phi(\mathcal{D}) = \sup_{\theta \in \Theta} \mathbb{E}_{(x,y)}[\ell(f(x;\theta), y)] - L_q(\theta; \mathcal{D})$. In this proof, our objective is to provide the upper bound on $\Phi(\mathcal{D})$ by using McDiarmid's inequality. To apply McDiarmid's inequality to $\Phi(\mathcal{D})$, we first show that $\Phi(\mathcal{D})$ satisfies the remaining condition of McDiarmid's inequality. Let $\mathcal{D}$ and $\mathcal{D}'$ be two datasets differing by exactly one point of an arbitrary index i_0; i.e., $\mathcal{D}_i = \mathcal{D}'_i$ for all $i \neq i_0$ and $\mathcal{D}_{i_0} \neq \mathcal{D}'_{i_0}$. Then, we provide an upper bound on $\Phi(\mathcal{D}') - \Phi(\mathcal{D})$ as follows:

$$
\begin{aligned}
\Phi(\mathcal{D}') - \Phi(\mathcal{D}) &\leq \sup_{\theta \in \Theta} L_q(\theta; \mathcal{D}) - L_q(\theta; \mathcal{D}'). \\
&= \sup_{\theta \in \Theta} \frac{1}{q} \sum_{j=1}^{n} \gamma_j (L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')) \\
&\leq \sup_{\theta \in \Theta} \frac{1}{q} \sum_{j=1}^{n} |\gamma_j| |L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')| \\
&\leq \sup_{\theta \in \Theta} \frac{1}{q} \frac{s}{n} \sum_{j=1}^{n} |L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')|
\end{aligned}
$$

where the first line follows the property of the supremum, $\sup(a) - \sup(b) \leq \sup(a - b)$, the second line follows the definition of L_q, and the last line follows Proposition 2 ($|\gamma_j| \leq \frac{s}{n}$).

We now bound the last term $\sum_{j=1}^{n} |L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')|$. This requires a careful examination because $|L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')| \neq 0$ for more than one index j (although $\mathcal{D}$ and $\mathcal{D}'$ differ only by exactly one point). This is because it is possible to have $(j; \mathcal{D}) \neq (j; \mathcal{D}')$ for many indexes j where $(j; \mathcal{D}) = (j)$ in $L_{(j)}(\theta; \mathcal{D})$ and $(j; \mathcal{D}') = (j)$ in $L_{(j)}(\theta; \mathcal{D}')$. To analyze this effect, we now conduct case analysis. Define $l(i; \mathcal{D})$ such that $(j) = i$ where $j = l(i; \mathcal{D})$; i.e., $L_i(\theta; \mathcal{D}) = L_{(l(i;\mathcal{D}))}(\theta; \mathcal{D})$.

Consider the case where $l(i_0; \mathcal{D}') \geq l(i_0; \mathcal{D})$. Let $j_1 = l(i_0; \mathcal{D})$ and $j_2 = l(i_0; \mathcal{D}')$. Then,

$$
\begin{aligned}
\sum_{j=1}^{n} |L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')| &= \sum_{j=j_1}^{j_2-1} |L_{(j)}(\theta; \mathcal{D}) - L_{(j)}(\theta; \mathcal{D}')| + |L_{(j_2)}(\theta; \mathcal{D}) - L_{(j_2)}(\theta; \mathcal{D}')| \\
&= \sum_{j=j_1}^{j_2-1} |L_{(j)}(\theta; \mathcal{D}) - L_{(j+1)}(\theta; \mathcal{D})| + |L_{(j_2)}(\theta; \mathcal{D}) - L_{(j_2)}(\theta; \mathcal{D}')| \\
&= \sum_{j=j_1}^{j_2-1} (L_{(j)}(\theta; \mathcal{D}) - L_{(j+1)}(\theta; \mathcal{D})) + L_{(j_2)}(\theta; \mathcal{D}) - L_{(j_2)}(\theta; \mathcal{D}') \\
&= L_{(j_1)}(\theta; \mathcal{D}) - L_{(j_2)}(\theta; \mathcal{D}') \\
&\leq M,
\end{aligned}
$$

where the first line uses the fact that $j_2 = l(i_0; \mathcal{D}') \geq l(i_0; \mathcal{D}) = j_1$ where i_0 is the index of samples differing in $\mathcal{D}$ and $\mathcal{D}'$. The second line follows the equality $(j; \mathcal{D}') = (j + 1; \mathcal{D})$ from j_1 to $j_2 - 1$ in this case. The third line follows the definition of the ordering of the indexes. The fourth line follows the cancellations of the terms from the third line.

Consider the case where $l(i_0;\mathcal{D}') < l(i_0;\mathcal{D})$. Let $j_1 = l(i_0;\mathcal{D}')$ and $j_2 = l(i_0;\mathcal{D})$. Then,

$$\begin{aligned}
\sum_{j=1}^{n} |L_{(j)}(\theta;\mathcal{D}) - L_{(j)}(\theta;\mathcal{D}')| &= |L_{(j_1)}(\theta;\mathcal{D}) - L_{(j_1)}(\theta;\mathcal{D}')| + \sum_{j=j_1+1}^{j_2} |L_{(j)}(\theta;\mathcal{D}) - L_{(j)}(\theta;\mathcal{D}')| \\
&= |L_{(j_1)}(\theta;\mathcal{D}) - L_{(j_1)}(\theta;\mathcal{D}')| + \sum_{j=j_1+1}^{j_2} |L_{(j)}(\theta;\mathcal{D}) - L_{(j-1)}(\theta;\mathcal{D})| \\
&= L_{(j_1)}(\theta;\mathcal{D}) - L_{(j_1)}(\theta;\mathcal{D}') + \sum_{j=j_1+1}^{j_2} (L_{(j)}(\theta;\mathcal{D}) - L_{(j-1)}(\theta;\mathcal{D})) \\
&= L_{(j_1)}(\theta;\mathcal{D}') - L_{(j_2)}(\theta;\mathcal{D}) \\
&\leq M.
\end{aligned}$$

where the first line uses the fact that $j_1 = l(i_0;\mathcal{D}') < l(i_0;\mathcal{D}) = j_2$ where i_0 is the index of samples differing in $\mathcal{D}$ and $\mathcal{D}'$. The second line follows the equality $(j;\mathcal{D}') = (j-1;\mathcal{D})$ from j_1+1 to j_2 in this case. The third line follows the definition of the ordering of the indexes. The fourth line follows the cancellations of the terms from the third line.

Therefore, in both cases of $l(i_0;\mathcal{D}') \geq l(i_0;\mathcal{D})$ and $l(i_0;\mathcal{D}') < l(i_0;\mathcal{D})$, we have that

$$\Phi(\mathcal{D}') - \Phi(\mathcal{D}) \leq \frac{s}{q}\frac{M}{n}.$$

Similarly, $\Phi(\mathcal{D}) - \Phi(\mathcal{D}') \leq \frac{s}{q}\frac{M}{n}$, and hence $|\Phi(\mathcal{D}) - \Phi(\mathcal{D}')| \leq \frac{s}{q}\frac{M}{n}$. Thus, by McDiarmid's inequality, for any $\delta > 0$, with probability at least $1-\delta$,

$$\Phi(\mathcal{D}) \leq \mathbb{E}_{\bar{\mathcal{D}}}[\Phi(\bar{\mathcal{D}})] + \frac{Ms}{q}\sqrt{\frac{\ln(1/\delta)}{2n}}.$$

Moreover, since

$$\sum_{i=1}^{n} r_i(\theta;\mathcal{D}) L_i(\theta;\mathcal{D}) = \sum_{j=1}^{n} \gamma_j \sum_{i=1}^{n} \mathbb{1}\{i=(j)\} L_i(\theta;\mathcal{D}) = \sum_{j=1}^{n} \gamma_j L_{(j)}(\theta;\mathcal{D}),$$

we have that

$$L_q(\theta;\mathcal{D}) = \frac{1}{q}\sum_{i=1}^{n} r_i(\theta;\mathcal{D}) L_i(\theta;\mathcal{D}) + R(\theta).$$

Therefore,

$$\begin{aligned}
&\mathbb{E}_{\bar{\mathcal{D}}}[\Phi(\bar{\mathcal{D}})] \\
&= \mathbb{E}_{\bar{\mathcal{D}}}\left[\sup_{\theta\in\Theta} \mathbb{E}_{(\bar{x}',\bar{y}')}[\ell(f(\bar{x}';\theta),\bar{y}')] - L(\theta;\bar{\mathcal{D}}) + L(\theta;\bar{\mathcal{D}}) - L_q(\theta;\bar{\mathcal{D}})\right] \\
&\leq \mathbb{E}_{\bar{\mathcal{D}}}\left[\sup_{\theta\in\Theta} \mathbb{E}_{(\bar{x}',\bar{y}')}[\ell(f(\bar{x}';\theta),\bar{y}')] - L(\theta;\bar{\mathcal{D}})\right] - \mathcal{Q}_n(\Theta;s,q) \\
&\leq \mathbb{E}_{\bar{\mathcal{D}},\bar{\mathcal{D}}'}\left[\sup_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n}(\ell(f(\bar{x}_i';\theta),\bar{y}_i') - \ell(f(\bar{x}_i;\theta),\bar{y}_i))\right] - \mathcal{Q}_n(\Theta;s,q) \\
&\leq \mathbb{E}_{\xi,\bar{\mathcal{D}},\bar{\mathcal{D}}'}\left[\sup_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n}\xi_i(\ell(f(\bar{x}_i';\theta),\bar{y}_i') - \ell(f(\bar{x}_i;\theta),\bar{y}_i))\right] - \mathcal{Q}_n(\Theta;s,q) \\
&\leq 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n(\Theta;s,q).
\end{aligned}$$

where the third line and the last line follow the subadditivity of supremum, the forth line follows the Jensen's inequality and the convexity of the supremum, the fifth line follows that for each $\xi_i \in \{-1,+1\}$, the distribution

of each term $\xi_i(\ell(f(\bar{x}'_i;\theta),\bar{y}'_i) - \ell(f(\bar{x}_i;\theta),\bar{y}_i))$ is the distribution of $(\ell(f(\bar{x}'_i;\theta),\bar{y}'_i) - \ell(f(\bar{x}_i;\theta),\bar{y}_i))$ since $\bar{\mathcal{D}}$ and $\bar{\mathcal{D}}'$ are drawn iid with the same distribution. Therefore, for any $\delta > 0$, with probability at least $1-\delta$,

$$\Phi(\mathcal{D}) \leq 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n(\Theta; s, q) + \frac{Ms}{q}\sqrt{\frac{\ln(1/\delta)}{2n}}.$$

□

B Additional discussion

The subset Θ in Theorem 3 characterizes the hypothesis space that is $\{x \mapsto f(x;\theta) : \theta \in \Theta\}$. An important subtlety here is that given a parameterized model f, one can apply Theorem 3 to a subset Θ that depends on an algorithm and a distribution (but not directly on a dataset) such as $\Theta = \{\theta \in \mathbb{R}^{d_y} : (\exists \mathcal{D} \in A)[\theta \text{ is the possible output of ordered SGD given } (f, \mathcal{D})]\}$ where A is a fixed set of the training datasets such that $\mathcal{D} \in A$ with high probability. Thus, even for the exact same model f and problem setting, Theorem 3 might provide non-vacuous bounds for some choices of Θ but not for other choices of Θ.

Moreover, we can easily obtain data-dependent bounds from Theorem 3 by repeatedly applying Theorem 3 to several subsets Θ and taking an union bound. For example, given a sequence $(\Theta_k)_{k\in\mathbb{N}^+}$, by applying Theorem 3 to each Θ_k with $\delta = \delta' \frac{6}{\pi^2 k^2}$ (for each k) and by taking a union bound over all $k \in \mathbb{N}^+$, the following statement holds: for any $\delta' > 0$, with probability at least $1-\delta'$ over an iid draw of n examples $\mathcal{D} = ((x_i, y_i))_{i=1}^n$, we have that for all $k \in \mathbb{N}^+$ and $\theta \in \Theta_k$,

$$\mathbb{E}_{(x,y)}[\ell(f(x;\theta),y)] \leq L_q(\theta;\mathcal{D}) + 2\mathfrak{R}_n(\Theta_k) + \frac{Ms}{q}\sqrt{\frac{\ln(\pi^2 k^2/6\delta')}{2n}} - \mathcal{Q}_n(\Theta_k; s, q).$$

For example, let us choose $\Theta_k = \{\theta \in \mathbb{R}^{d_y} : \|\theta\| \leq c_k\}$ with some constants $c_1 < c_2 < \cdots$. Then, when we obtain a $\hat{\theta}_q$ after training based on a particular training dataset $\mathcal{D}$ such that $c_{\bar{k}-1} < \|\hat{\theta}_q\| \leq c_{\bar{k}}$ for some $\bar{k}$, we can conclude the following: with probability at least $1-\delta'$, $\mathbb{E}_{(x,y)}[\ell(f(x;\theta),y)] \leq L_q(\hat{\theta}_q;\mathcal{D}) + 2\mathfrak{R}_n(\Theta_{\bar{k}}) + \frac{Ms}{q}\sqrt{\frac{\ln(\pi k^2/6\delta')}{2n}} - \mathcal{Q}_n(\Theta_{\bar{k}}; s, q)$. This is data-dependent in the sense that $\Theta_{\bar{k}}$ is selected in the data-dependent manner from $(\Theta_k)_{k\in\mathbb{N}^+}$. This is in contrast to the fact that as logically indicated in the theorem statement, one cannot directly apply Theorem 3 to a single subset Θ that directly depends on training dataset; e.g., one *cannot* apply Theorem 3 to a singleton set $\hat{\Theta}(\mathcal{D}) = \{\hat{\theta}(\mathcal{D})\}$ where $\hat{\theta}(\mathcal{D})$ is the output of training given $\mathcal{D}$.

C Additional experimental results and details

C.1 Additional results

Wall-clock time. Table 4 summarises the wall-clock time values (in seconds) of mini-batch SGD and ordered SGD. The wall-clock time was computed with identical, independent, and freed GPUs for fair comparison. The wall-clock time measures the time of the whole computations, including the extra computation of finding a set Q of top-q samples in S in term of loss values. As it can be seen, the extra computation of finding a set Q of top-q samples is generally negligible. Furthermore, for larger scale problems, ordered SGD tends to be faster per epoch because of the computational saving of not using the full mini-batch for the backpropagation computation.

Effect of different learning rates and mini-batch sizes. Figures 5 and 6 show the results with different learning rates and mini-batch sizes. Both use the same setting as that for CIFAR-10 with no data augmentation in others results shown in Table 1 and Figure 3. Figures 5 and 6 consistently show improvement of ordered SGD over mini-batch SGD for all learning rates and mini-batch sizes.

Behaviors with different datasets. Figure 7 shows the behaviors of mini-batch SGD vs ordered SGD. As it can be seen, ordered SGD generally improved mini-batch SGD in terms of test errors. With data argumentation, we also tried linear logistic regression for the Semeion dataset, and obtained the mean test errors of 19.11 for mini-batch SGD and 16.54 for ordered SGD (the standard deviations were 1.48 and 1.24); i.e., ordered SGD

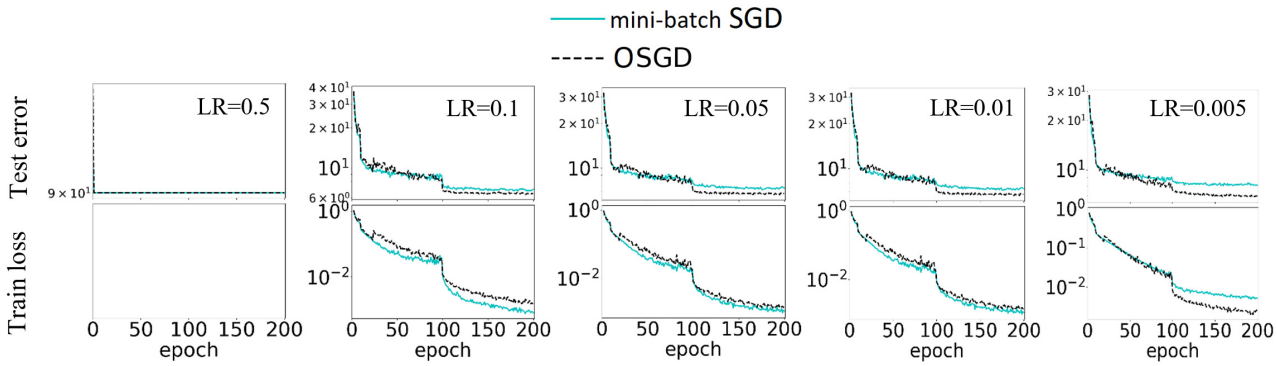


Figure 5: Test error and training loss (in log scales) versus the number of epoch with CIFAR-10 and no data augmentation by using different learning rates (LRs). The plotted values indicate the mean values over 10 random trials. The training loss values of LR=0.5 were 'nan' for both methods.

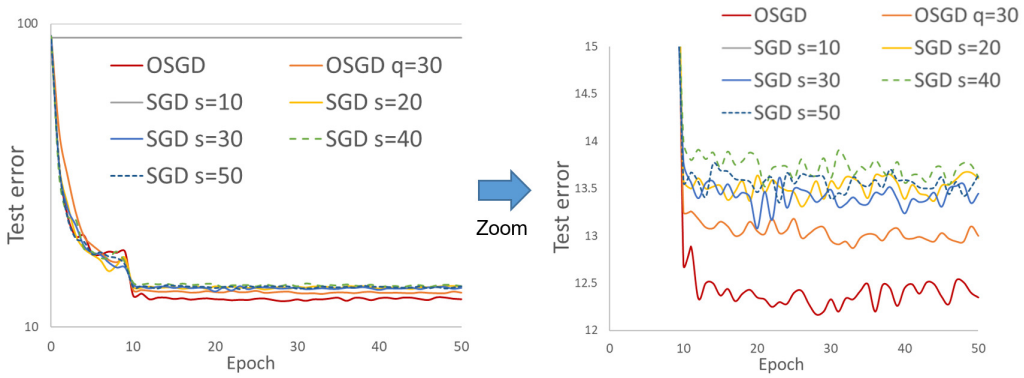


Figure 6: Test error versus the number of epoch with CIFAR-10 and no data augmentation by using different mini-batch sizes s.

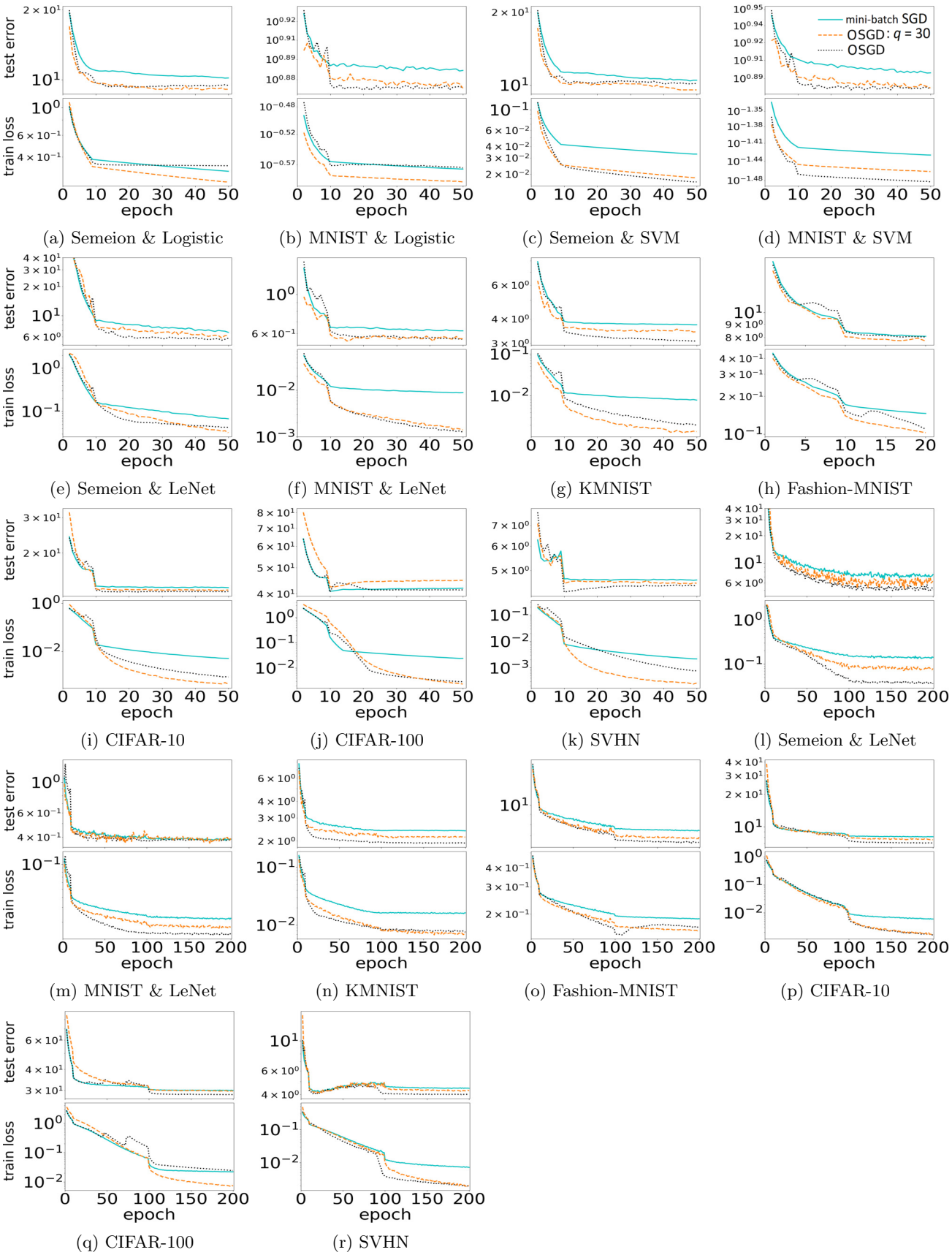
improved over mini-batch SGD, but the mean test errors without data-augmentation were better for both mini-batch SGD and ordered SGD. This is because the data augmentation made it difficult to fit the augmented training dataset with linear models.

Effect of different values of q. Figure 8 shows the behaviors of mini-batch SGD vs ordered SGD with different q values. In the figure, label 'ordered SGD' corresponds to ordered SGD with the fixed adaptive rule, and other labels (e.g., 'ordered SGD: $q = 10$') corresponds to ordered SGD with the fixed value of q over the whole training procedure (e.g., with $q = 10$). All experiments in the figure were conducted with data augmentations. PreActResNet18 was used for CIFAR-10, while LeNet was used for other datasets. As it can be seen in Figure 8, ordered SGD generally improved the test errors of mini-batch SGD, even with fixed q values. When the value of q is fixed to be small as in $q = 10$, the small q value can be effective during the latter stage of training (e.g., Figure 8 b) while the training can be inefficient during the initial stage of training (e.g., Figure 8 c).

Results with ordered Adam. Table 5 compares the testing performance of ordered Adam and (standard) Adam for different models and datasets. The table reports the mean and the standard deviation of test errors (i.e., 100 × the average of 0-1 losses on test dataset) over 10 random experiments with different random seeds. The procedures of ordered Adam follow those of Adam except the additional sample strategy (line 3 - 4 of Algorithm 1). Table 5 shows that ordered Adam improved Adam for all settings, except CIFAR-10 with data augmentation. For CIFAR-10 with data augmentation, ordered SGD preformed the best among mini-batch SGD, Adam, ordered SGD, and ordered Adam, as it can be seen in Tables 1 and 5.

Table 4: Average wall-clock time (seconds) per epoch.

Data Aug	Datasets	Model	mini-batch SGD	ordered SGD	difference
No	Semeion	Logistic model	0.15 (0.01)	0.15 (0.01)	0.00
No	MNIST	Logistic model	7.16 (0.27)	7.32 (0.24)	-0.16
No	Semeion	SVM	0.17 (0.01)	0.17 (0.01)	0.00
No	MNIST	SVM	8.60 (0.31)	8.72 (0.29)	-0.12
No	Semeion	LeNet	0.18 (0.01)	0.18 (0.01)	0.00
No	MNIST	LeNet	9.00 (0.34)	9.12 (0.27)	-0.12
No	KMNIST	LeNet	9.23 (0.33)	9.04 (0.55)	0.19
No	Fashion-MNIST	LeNet	8.56 (0.48)	9.45 (0.31)	-0.90
No	CIFAR-10	PreActResNet18	45.55 (0.47)	43.72 (0.93)	1.82
No	CIFAR-100	PreActResNet18	46.83 (0.90)	43.95 (1.03)	2.89
No	SVHN	PreActResNet18	71.95 (1.40)	66.94 (1.67)	5.01
Yes	Semeion	LeNet	0.28 (0.02)	0.28 (0.02)	0.00
Yes	MNIST	LeNet	14.44 (0.54)	14.77 (0.41)	-0.32
Yes	KMNIST	LeNet	12.17 (0.33)	11.42 (0.29)	0.75
Yes	Fashion-MNIST	LeNet	12.23 (0.40)	12.38 (0.37)	-0.14
Yes	CIFAR-10	PreActResNet18	48.18 (0.58)	46.40 (0.97)	1.78
Yes	CIFAR-100	PreActResNet18	47.37 (0.84)	44.74 (0.91)	2.63
Yes	SVHN	PreActResNet18	72.29 (1.23)	67.95 (1.54)	4.34



(a) Semeion & Logistic (b) MNIST & Logistic (c) Semeion & SVM (d) MNIST & SVM

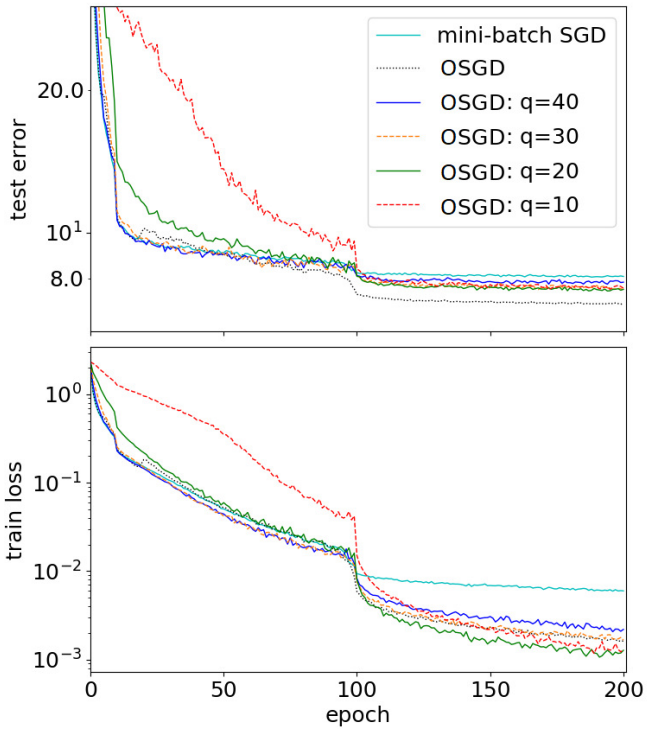
(e) Semeion & LeNet (f) MNIST & LeNet (g) KMNIST (h) Fashion-MNIST

(i) CIFAR-10 (j) CIFAR-100 (k) SVHN (l) Semeion & LeNet

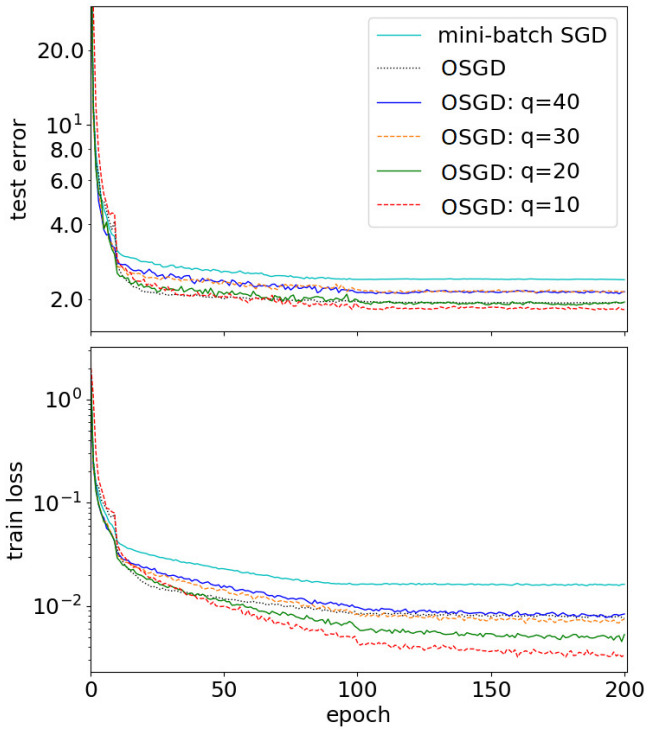
(m) MNIST & LeNet (n) KMNIST (o) Fashion-MNIST (p) CIFAR-10

(q) CIFAR-100 (r) SVHN

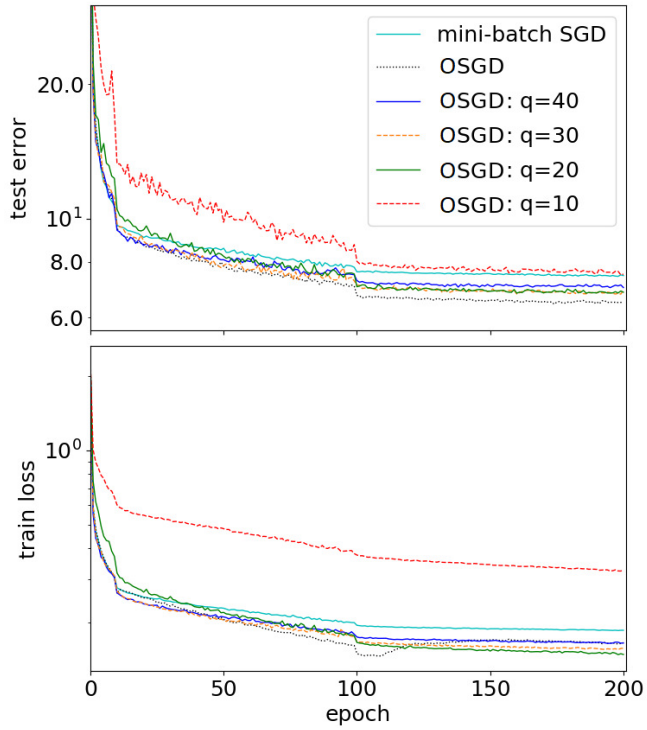
Figure 7: Test error and training loss (in log scales) versus epoch for all experiments with mini-batch SGD and ordered SGD. These are without data augmentation in subfigures (a)-(k), and with data augmentation in subfigures (l)-(r). The plotted values are the mean values over ten random trials.



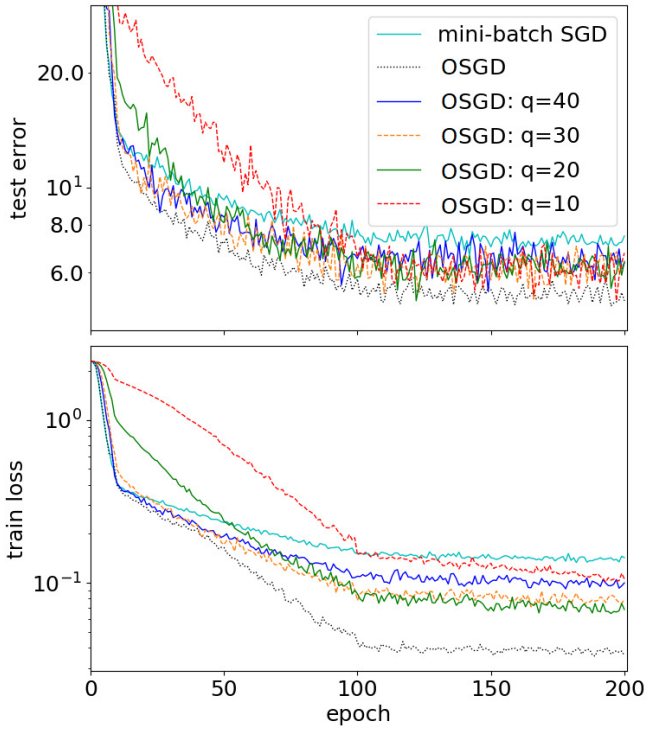
(a) CIFAR-10



(b) KMNIST



(c) Fashion-MNIST



(d) Semeion

Figure 8: Effect of different values of q.

Table 5: Test errors (%) of Adam and ordered Adam. The last column labeled "Improve" shows relative improvements (%) from Adam to ordered Adam. In the other columns, the numbers indicate the mean test errors (and standard deviations in parentheses) over ten random trials. The first column shows 'No' for no data augmentation, and 'Yes' for data augmentation.

Data Aug	Datasets	Model	Adam	ordered Adam	Improve
No	Semeion	Logistic model	12.12 (0.71)	10.37 (0.77)	14.46
No	MNIST	Logistic model	7.34 (0.03)	7.20 (0.03)	1.97
No	Semeion	SVM	11.45 (0.90)	10.91 (0.86)	4.71
No	MNIST	SVM	7.53 (0.03)	7.43 (0.02)	1.38
No	Semeion	LeNet	6.21 (0.64)	5.75 (0.42)	7.34
No	MNIST	LeNet	0.70 (0.04)	0.63 (0.04)	10.07
No	KMNIST	LeNet	3.14 (0.13)	3.13 (0.14)	0.60
No	Fashion-MNIST	LeNet	7.79 (0.17)	7.79 (0.21)	0.01
No	CIFAR-10	PreActResNet18	13.21 (0.42)	12.98 (0.27)	1.68
No	CIFAR-100	PreActResNet18	45.33 (0.89)	44.42 (0.72)	2.01
No	SVHN	PreActResNet18	4.72 (0.12)	4.64 (0.09)	1.52
Yes	Semeion	LeNet	5.80 (0.85)	5.70 (0.60)	1.74
Yes	MNIST	LeNet	0.45 (0.05)	0.44 (0.02)	3.10
Yes	KMNIST	LeNet	2.01 (0.08)	1.94 (0.16)	3.49
Yes	Fashion-MNIST	LeNet	6.61 (0.14)	6.56 (0.14)	0.82
Yes	CIFAR-10	PreActResNet18	7.92 (0.28)	8.03 (0.13)	-1.39
Yes	CIFAR-100	PreActResNet18	32.24 (0.52)	32.03 (0.52)	0.65
Yes	SVHN	PreActResNet18	4.42 (0.12)	4.19 (0.11)	5.29

C.2 Additional details

For all experiments, mini-batch SGD and ordered SGD (as well as Adam and ordered Adam) were run with the same machine and the same PyTorch codes except a single-line modification:

- `loss = torch.mean(loss)` for mini-batch SGD and Adam
- `loss = torch.mean(torch.topk(loss, min(q, s), sorted=False, dim=0)[0])` for ordered SGD and ordered Adam.

For 2-D illustrations in Figure 1. We used the (binary) cross entropy loss, $s = 100$, and 2 dimensional synthetic datasets with $n = 200$ in Figures 1a–1b and $n = 1000$ in Figures 1c–1d. The artificial neural network (ANN) used in Figures 1c and 1d is a fully-connected feedforward neural network with rectified linear units (ReLUs) and three hidden layers, where each hidden layer contained 20 neurons in Figures 1c and 10 neurons in Figures 1d.

For other numerical results. For mixup and random erasing, we used the same setting as in the corresponding previous papers (Zhong et al., 2017; Verma et al., 2019). For others, we divided the learning rate by 10 at the beginning of 10th epoch for all experiments (with and without data augmentation), and of 100th epoch for those with data augmentation. With $y \in \{1, \dots, d_y\}$, we used the cross entropy loss $\ell(a, y) = -\log \frac{\exp(a_y)}{\sum_{k'} \exp(a_{k'})}$ for neural networks as well as multinomial logistic models, and a multiclass hinge loss $\ell(a, y) = \sum_{k \neq y} \max(0, 1 + a_k - a_y)$ for SVMs (Weston et al., 1999). For the variant of LeNet, we used the following architecture with five layers (three hidden layers):

1. Input layer
2. Convolutional layer with 64 5×5 filters, followed by max pooling of size of 2 by 2 and ReLU.
3. Convolutional layer with 64 5×5 filters, followed by max pooling of size of 2 by 2 and ReLU.
4. Fully connected layer with 1014 output units, followed by ReLU.
5. Fully connected layer with the number of output units being equal to the number of target classes.